

SQL Projects

Data Analysis



SQL PROJECT- MUSIC STORE DATA ANALYSIS

Question Set 1 -

1. Who is the senior most employee based on job title?
2. Which countries have the most Invoices?
3. What are top 3 values of total invoice?
4. Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money. Write a query that returns one city that has the highest sum of invoice totals. Return both the city name & sum of all invoice totals
5. Who is the best customer? The customer who has spent the most money will be declared the best customer. Write a query that returns the person who has spent the most money

Question Set 2 –

1. Write query to return the email, first name, last name, & Genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A
2. Let's invite the artists who have written the most rock music in our dataset. Write a query that returns the Artist name and total track count of the top 10 rock bands
3. Return all the track names that have a song length longer than the average song length. Return the Name and Milliseconds for each track. Order by the song length with the longest songs listed first

Question Set 3 –

1. Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent
2. We want to find out the most popular music Genre for each country. We determine the most popular genre as the genre with the highest amount of purchases. Write a query that returns each country along with the top Genre. For countries where the maximum number of purchases is shared return all Genres
3. Write a query that determines the customer that has spent the most on music for each country. Write a query that returns the country along with the top customer and how much they spent. For countries where the top amount spent is shared, provide all customers who spent this amount

SQL Code

```
1 -- Business Questions & Answers
2
3
4 -- Q1: Who is the senior most employee based on job title?
5
6 CREATE VIEW senior_most_employee AS
7 SELECT title, last_name, first_name
8 FROM employee
9 ORDER BY levels DESC
10 LIMIT 1;
11
12 -- Q2: Which countries have the most Invoices?
13
14 CREATE VIEW invoice_count_by_country AS
15 SELECT COUNT(*) AS c, billing_country
16 FROM invoice
17 GROUP BY billing_country
18 ORDER BY c DESC;
19
20 -- Q3: What are top 3 values of total invoice?
21
22 CREATE VIEW top_3_invoice_totals AS
23 SELECT total
24 FROM invoice
25 ORDER BY total DESC
26 LIMIT 3;
27
28 -- Q4: Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money.
29 -- Write a query that returns one city that has the highest sum of invoice totals.
30 -- Return both the city name & sum of all invoice totals
31
32 CREATE VIEW best_customer_city AS
33 SELECT billing_city, SUM(total) AS InvoiceTotal
34 FROM invoice
35 GROUP BY billing_city
36 ORDER BY InvoiceTotal DESC
37 LIMIT 1;
38
39 -- Q5: Who is the best customer? The customer who has spent the most money will be declared the best customer.
40 -- Write a query that returns the person who has spent the most money.
41
42 CREATE VIEW best_customer AS
43 SELECT customer.customer_id, first_name, last_name, SUM(total) AS total_spending
44 FROM customer
45 JOIN invoice ON customer.customer_id = invoice.customer_id
46 GROUP BY customer.customer_id, first_name, last_name
47 ORDER BY total_spending DESC
48 LIMIT 1;
49
50 -- 2nd phase questions:
```

```
51
52 Q1: Write query to return the email, first name, last name, & Genre of all Rock Music listeners.
53   Return your list ordered alphabetically by email starting with A.
54
55 -- Method 1:
56
57 CREATE VIEW rock_music_listeners_method1 AS
58   SELECT DISTINCT email, first_name, last_name
59   FROM customer
60   JOIN invoice ON customer.customer_id = invoice.customer_id
61   JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
62   WHERE invoice_line.track_id IN (
63     SELECT track_id
64     FROM track
65     JOIN genre ON track.genre_id = genre.genre_id
66     WHERE genre.name LIKE 'Rock'
67   )
68   ORDER BY email;
69
70 -- Method 2:
71
72 CREATE VIEW rock_listeners_method2 AS
73   SELECT DISTINCT
74     customer.email,
75     customer.first_name,
76     customer.last_name,
77     genre.name AS genre_name
78   FROM customer
79   JOIN invoice
80     ON invoice.customer_id = customer.customer_id
81   JOIN invoice_line
82     ON invoice_line.invoice_id = invoice.invoice_id
83   JOIN track
84     ON track.track_id = invoice_line.track_id
85   JOIN genre
86     ON genre.genre_id = track.genre_id
87   WHERE genre.name = 'Rock'
88   ORDER BY customer.email;
89
90
91 -- Q2: Let's invite the artists who have written the most rock music in our database.
92 -- -- Write a query that returns the Artist name and total track count of the top 10 rock bands.
93
94 CREATE VIEW top_rock_artists AS
95   SELECT artist.artist_id, artist.name, COUNT(artist.artist_id) AS number_of_songs
96   FROM track
97   JOIN album ON album.album_id = track.album_id
98   JOIN artist ON artist.artist_id = album.artist_id
99   JOIN genre ON genre.genre_id = track.genre_id
100  WHERE genre.name LIKE 'Rock'
101  GROUP BY artist.artist_id, artist.name
102  ORDER BY number_of_songs DESC
```

```
103    LIMIT 10;
104
105    -- Q3: Return all the track names that have a song length longer than the average
106    -- song length.
107    -- Return the Name and Milliseconds for each track. Order by the song length with t
108    -- he longest songs listed first.
109
110    CREATE VIEW tracks_longer_than_average AS
111        SELECT name, milliseconds
112        FROM track
113        WHERE milliseconds > (
114            SELECT AVG(milliseconds)
115            FROM track
116        )
117        ORDER BY milliseconds DESC;
118
119
120    -- 3rd Phase Questions
121
122    -- Q1: Find how much amount spent by each customer on artists? Write a query to return
123    -- customer name, artist name and total spent */
124
125    -- Steps to Solve: First, find which artist has earned the most according to the In
126    -- voiceLines. Now use this artist to find
127    -- which customer spent the most on this artist. For this query, you will need to u
128    -- se the Invoice, InvoiceLine, Track, Customer,
129    -- Album, and Artist tables. Note, this one is tricky because the Total spent in th
130    -- e Invoice table might not be on a single product,
131    -- so you need to use the InvoiceLine table to find out how many of each product wa
132    -- s purchased, and then multiply this by the price
133
134    -- for each artist.
135
136    CREATE VIEW customer_spending_on_top_artist AS
137        SELECT
138            c.customer_id,
139            c.first_name,
140            c.last_name,
141            a.name AS artist_name,
142            SUM(il.unit_price * il.quantity) AS amount_spent
143        FROM invoice i
144        JOIN customer c ON c.customer_id = i.customer_id
145        JOIN invoice_line il ON il.invoice_id = i.invoice_id
146        JOIN track t ON t.track_id = il.track_id
147        JOIN album alb ON alb.album_id = t.album_id
148        JOIN artist a ON a.artist_id = alb.artist_id
149        WHERE a.artist_id = (
150            SELECT artist.artist_id
151            FROM invoice_line
152            JOIN track ON track.track_id = invoice_line.track_id
153            JOIN album ON album.album_id = track.album_id
154            JOIN artist ON artist.artist_id = album.artist_id
155            GROUP BY artist.artist_id, artist.name
156            ORDER BY SUM(invoice_line.unit_price * invoice_line.quantity) DESC
157            LIMIT 1
158        )
```

```

151 GROUP BY c.customer_id, c.first_name, c.last_name, a.name
152 ORDER BY amount_spent DESC;
153
154
155 -- Q2: We want to find out the most popular music Genre for each country. We determine the most popular genre as the genre
156 -- with the highest amount of purchases. Write a query that returns each country along with the top Genre. For countries where
157 -- the maximum number of purchases is shared return all Genres.
158
159 -- Steps to Solve: There are two parts in question- first most popular music genre and second need data at country level. */
160
161 -- Method 1:
162
163 CREATE VIEW top_genre_by_country AS
164 WITH popular_genre AS
165 (
166     SELECT COUNT(invoice_line.quantity) AS purchases, customer.country,
167             genre.name, genre.genre_id,
168             ROW_NUMBER() OVER(PARTITION BY customer.country
169                             ORDER BY COUNT(invoice_line.quantity) DESC) AS RowNo
170         FROM invoice_line
171     JOIN invoice ON invoice.invoice_id = invoice_line.invoice_id
172     JOIN customer ON customer.customer_id = invoice.customer_id
173     JOIN track ON track.track_id = invoice_line.track_id
174     JOIN genre ON genre.genre_id = track.genre_id
175     GROUP BY 2,3,4
176 )
177     SELECT * FROM popular_genre WHERE RowNo = 1;
178     select * from top_genre_by_country
179
180 -- Method 2:
181
182 CREATE VIEW top_genre_by_country_recursive AS
183 WITH RECURSIVE
184     sales_per_country AS(
185         SELECT COUNT(*) AS purchases_per_genre, customer.country, genre.name, genre.
186             genre_id
187             FROM invoice_line
188             JOIN invoice ON invoice.invoice_id = invoice_line.invoice_id
189             JOIN customer ON customer.customer_id = invoice.customer_id
190             JOIN track ON track.track_id = invoice_line.track_id
191             JOIN genre ON genre.genre_id = track.genre_id
192             GROUP BY 2,3,4
193     ),
194     max_genre_per_country AS (
195         SELECT MAX(purchases_per_genre) AS max_genre_number, country
196             FROM sales_per_country
197             GROUP BY 2
198     )
199     SELECT sales_per_country.*
200     FROM sales_per_country
201     JOIN max_genre_per_country
202     ON sales_per_country.country = max_genre_per_country.country

```

```
202 WHERE sales_per_country.purchases_per_genre = max_genre_per_country.max_genre_number;
r;
203 select * from top_genre_by_country_recursive
204
205
206 -- Q3: Write a query that determines the customer that has spent the most on music
for each country.
207 -- -- Write a query that returns the country along with the top customer and how mu
ch they spent.
208 -- -- For countries where the top amount spent is shared, provide all customers who
spent this amount.
209
210 -- Steps to Solve: Similar to the above question. There are two parts in question-
211 -- first find the most spent on music for each country and second filter the data f
or respective customers.
212
213 -- Method 1:
214
215 CREATE VIEW top_customer_by_country AS
216 WITH customer_with_country AS (
217     SELECT customer.customer_id, first_name, last_name, billing_country,
218             SUM(total) AS total_spending,
219             ROW_NUMBER() OVER(PARTITION BY billing_country ORDER BY SUM(total) DESC)
AS RowNo
220     FROM invoice
221     JOIN customer ON customer.customer_id = invoice.customer_id
222     GROUP BY 1,2,3,4
223 )
224     SELECT * FROM customer_with_country WHERE RowNo = 1;
225
226 -- Method 2:
227
228 CREATE VIEW top_customer_by_country_recursive AS
229 WITH
230     customer_with_country AS (
231         SELECT
232             customer.customer_id,
233             customer.first_name,
234             customer.last_name,
235             invoice.billing_country,
236             SUM(invoice.total) AS total_spending
237         FROM invoice
238         JOIN customer
239             ON customer.customer_id = invoice.customer_id
240         GROUP BY
241             customer.customer_id,
242             customer.first_name,
243             customer.last_name,
244             invoice.billing_country
245     ),
246
247     country_max_spending AS (
248         SELECT
249             billing_country,
250             MAX(total_spending) AS max_spending
```

```
251      FROM customer_with_country
252      GROUP BY billing_country
253  )
254
255  SELECT
256      cc.billing_country,
257      cc.total_spending,
258      cc.first_name,
259      cc.last_name,
260      cc.customer_id
261  FROM customer_with_country cc
262  JOIN country_max_spending ms
263      ON cc.billing_country = ms.billing_country
264      AND cc.total_spending = ms.max_spending
265  ORDER BY cc.billing_country;
266
267
268
269
270
271
272
273
274
```

Result Grid | Filter Rows: _____ | Export: Wrap Cell Content:

	customer_id	first_name	last_name	total_spending
▶	5	František	Wichterlová	144.54000000000002

Result Grid | Filter Rows: Export: Wrap Cell Content:

	billing_city	InvoiceTotal
▶	Prague	273.24000000000007

Result Grid | Filter Rows: Export: Wrap Cell Content:

	customer_id	first_name	last_name	artist_name	amount_spent
▶	54	Steve	Murray	AC/DC	17.82
	53	Phil	Hughes	AC/DC	10.89
	21	Kathy	Chase	AC/DC	10.89
	49	Stanisław	Wojciech	AC/DC	9.9
	1	Luís	Gonçalves	AC/DC	7.9200000000000001

customer_spending_on_top_artists

Result Grid

Filter Rows:

	c	billing_country
▶	131	USA
	76	Canada
	61	Brazil
	50	France
	41	Germany

invoice_count_by_country 18 ×

Result Grid | Filter Rows: _____ | Export: | Wrap Cell Content:

	email	first_name	last_name	genre_name
▶	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
	alero@uol.com.br	Alexandre	Rocha	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	bjorn.hansen@yahoo.no	Bjørn	Hansen	Rock
	camille.bernard@yahoo.fr	Camille	Bernard	Rock

rock_listeners_method2 19 x

Result Grid | Filter Rows: _____ | Export: | Wrap Cell Content:

	email	first_name	last_name
▶	aaronmitchell@yahoo.ca	Aaron	Mitchell
	alero@uol.com.br	Alexandre	Rocha
	astrid.gruber@apple.at	Astrid	Gruber
	bjorn.hansen@yahoo.no	BjÃ¸rn	Hansen
	camille.berr camille.bernard@yahoo.fr		Bernard

rock_music_listeners_method120 x

Result Grid

Filter Rows:

	title	last_name	first_name
▶	General Manager	Adams	Andrew

Result Grid



Filter Rows:

	total
▶	23.759999999999998
	19.8
	19.8

Result Grid | Filter Rows: Export: Wrap Cell Content:

	customer_id	first_name	last_name	billing_country	total_spending	RowNo
▶	56	Diego	Gutiérrez	Argentina	39.6	1
	55	Mark	Taylor	Australia	81.18	1
	7	Astrid	Gruber	Austria	69.3	1
	8	Daan	Peeters	Belgium	60.38999999999999	1
	1	Luís	Gonçalves	Brazil	108.89999999999998	1

top_customer_by_country 23 ×

Result Grid | Filter Rows: _____ | Export: | Wrap Cell Content:

	billing_country	total_spending	first_name	last_name	customer_id
▶	Argentina	39.6	Diego	Gutiérrez	56
	Australia	81.18	Mark	Taylor	55
	Austria	69.3	Astrid	Gruber	7
	Belg	Belgium	Daan	Peeters	8
	Brazil	108.8999999999998	Luís	Gonçalves	1

top_customer_by_country_reu...

Result Grid



Filter Rows:

Export:

	purchases	country	name	genre_id	RowNo
▶	1	Argentina	Rock	1	1
	18	Australia	Rock	1	1
	6	Austria	Australia	Rock	1
	5	Belgium		Rock	1
	26	Brazil		Rock	1

top_genre_by_country 25 x

Result Grid | Filter Rows:

Export:

	purchases_per_genre	country	name	genre_id
▶	26	Brazil	Rock	1
	6	Denmark	Rock	1
	23	Portugal	Rock	1
	7	Chile	Rock	1
	70	USA	Rock	1

top_genre_by_country_recursiv...

Result Grid



Filter Rows:

Export:



Wr

	artist_id	name	number_of_songs
▶	1	AC/DC	8
	3	Aerosmith	15
	8	Audioslave	14
	22	Led Zeppelin	14
	4	Alanis Morissette	13

top_rock_artists 27 ×

Output

Result Grid | Filter Rows: _____ | Export:

name	milliseconds
How Many More Times	711836
Advance Romance	677694
Sleeping Village	644571
You Shook Me(2)	619467
Talkin' 'Bout Women Obviously	589531

tracks_longer_than_average 28